UNIVERSITY OF CAMBRIDGE

Understanding and Improving Representation Learning in the Presence of Shortcuts

Batu El
Department of Computer Science and Technology
University of Cambridge, UK

Supervisors: Prof. Andreas Vlachos & Dr. Michalis Korakakis

Contents

Motivation

Problem

Existing Approaches

A New Perspective

Motivation

Motivation: Advances & Applications

Advances in Language Modeling

Transformer (Vaswani et al., 2017)

BERT (Devlin et al., 2018)*

GPT (Radford et al., 2018)

Applications on social media platforms:

1. Fact-check statements (Thorne et al., 2018) → Natural Language Inference
2. Detect harmful comments (Borkan et al., 2019) → Toxicity Detection

* We use BERT for our text classification tasks

Motivation: Accuracy & Reliability

Two notable challenges remain on the the ***reliability*** front

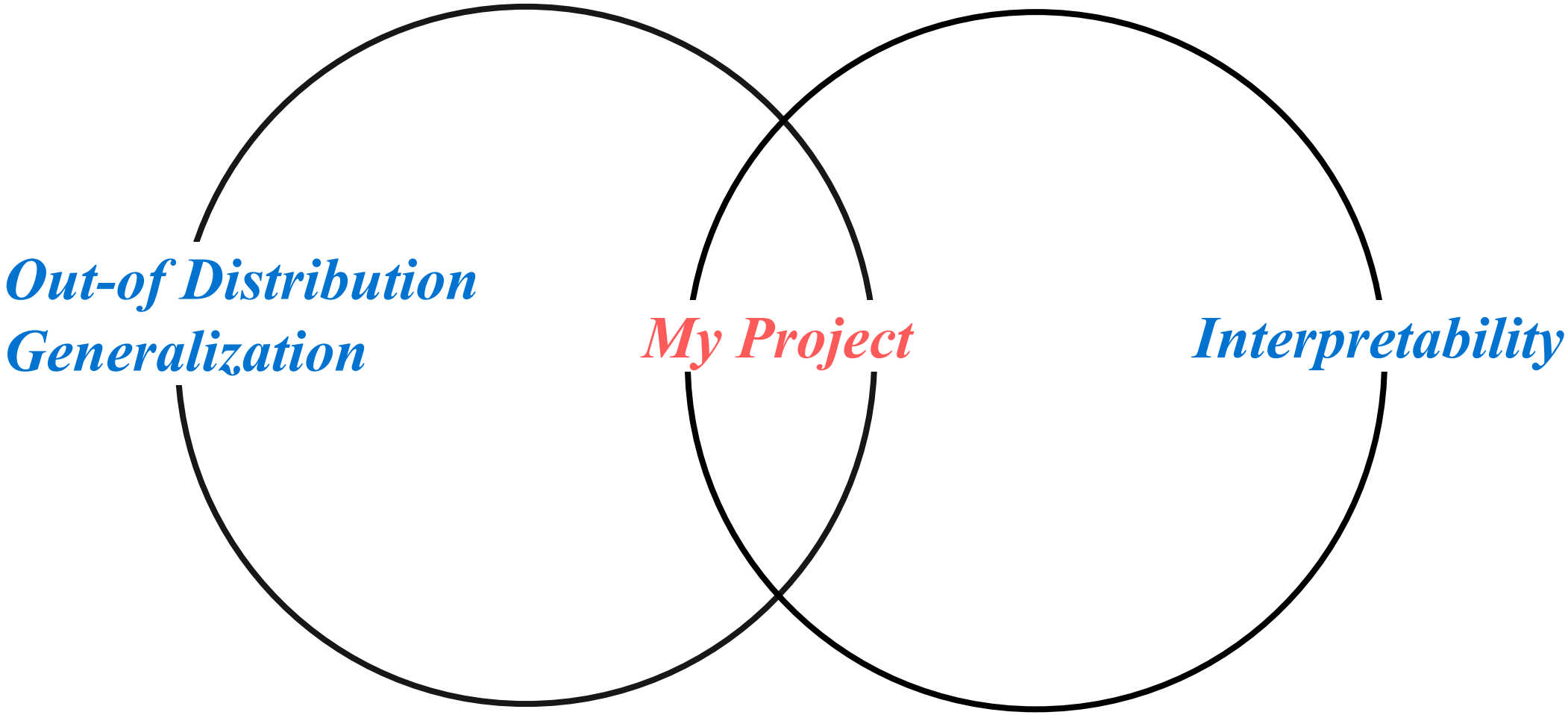
1. **Out-of-Distribution Generalization:**

Poor performance on specific data groups → Poor out-of-distribution generalization

2. **Interpretability:**

Opaque decision-making processes → Undermines safe deployment in sensitive domains.

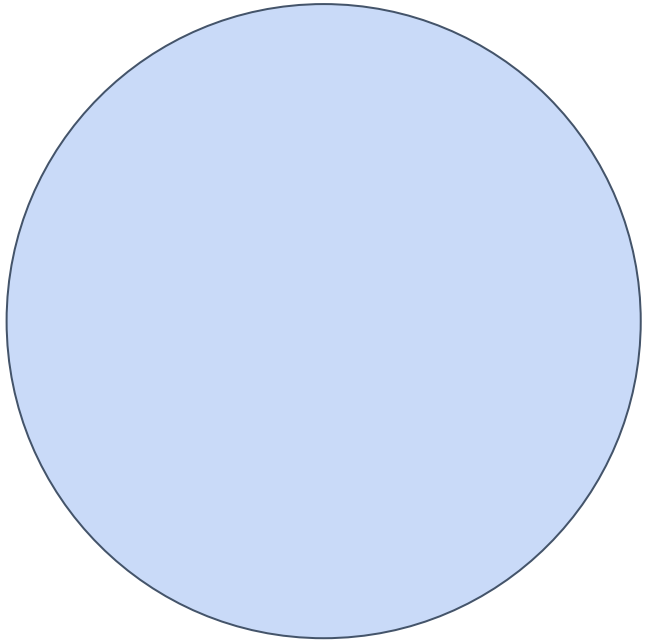
Motivation


Out-of Distribution Generalization
My Project
Interpretability

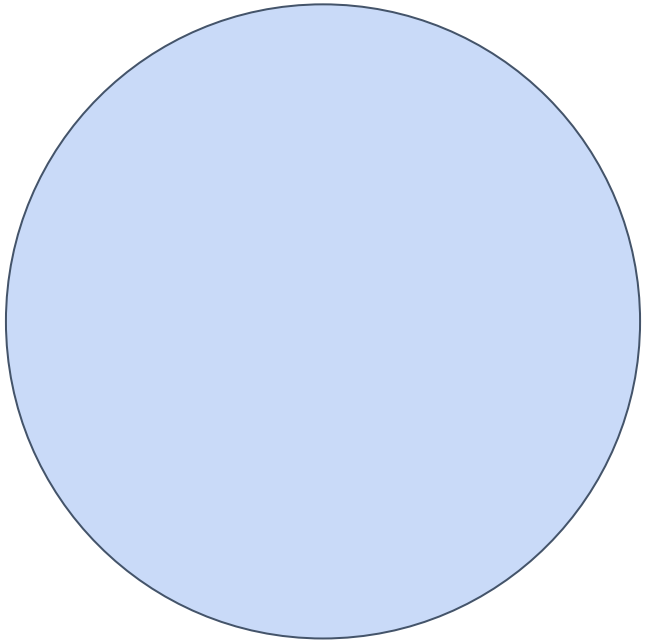
Problem

Problem: Evaluating Models

Training Split



IID Test Split



Generalization & Overfitting

Accuracy on Training Split:
75%

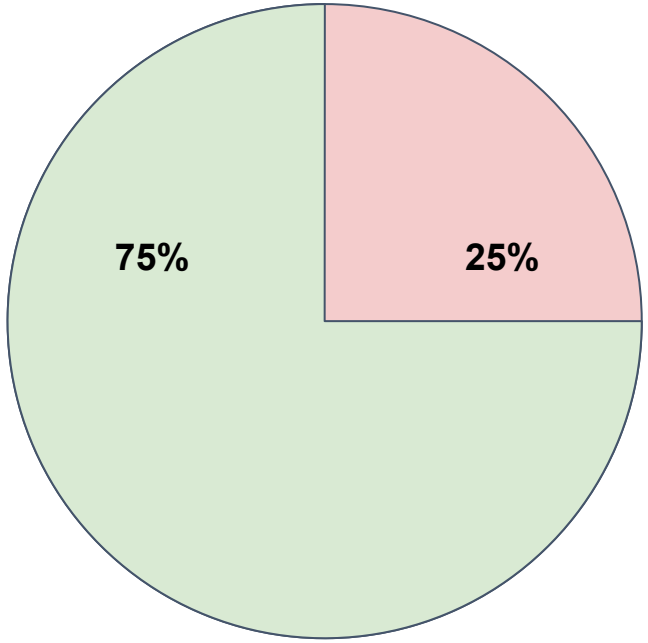
Accuracy on IID Test Split:
75%

Problem: Bad Performance on Certain Groups of Examples

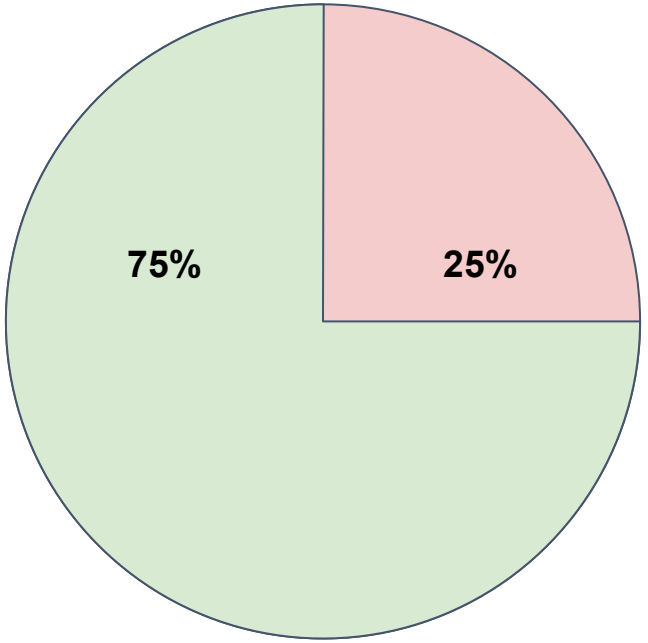
Data Group A *(100% Accuracy)*

Data Group B *(0% Accuracy)*

Training Split



IID Test Split



Accuracy on Training Split:
(0.75 * 100) + (0.25 * 0) = 75%

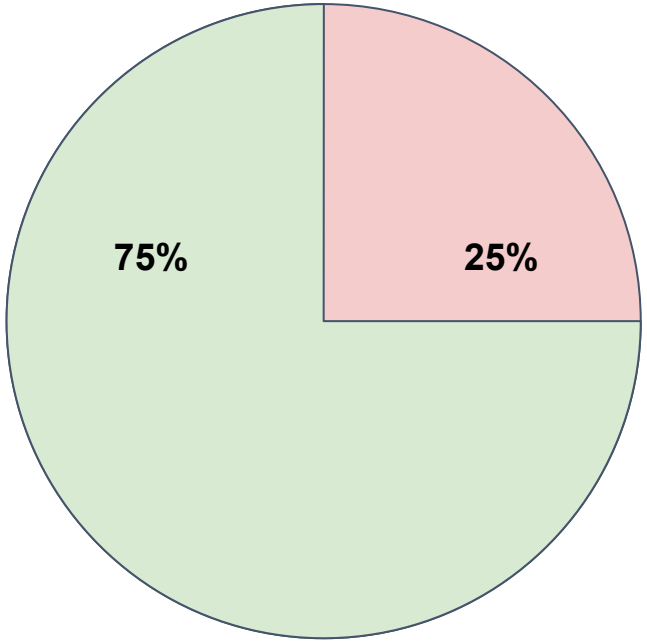
Accuracy on IID Test Split:
(0.75 * 100) + (0.25 * 0) = 75%

Problem: Bad Out-of-Distribution Performance

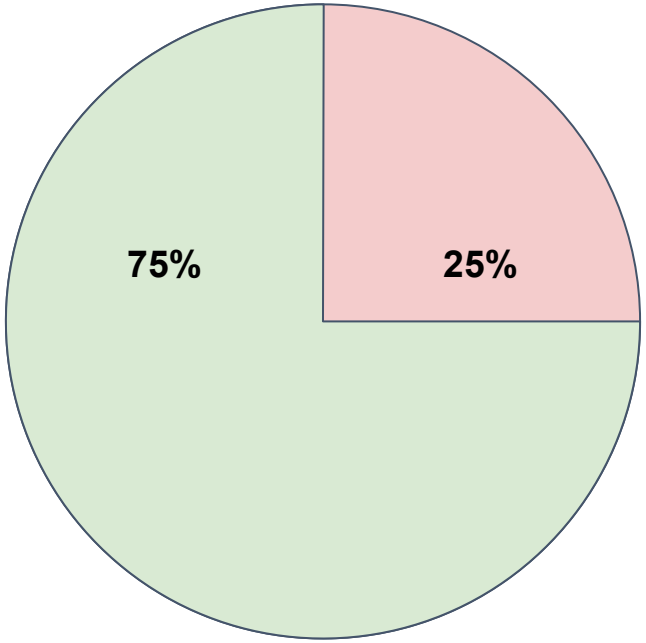
Data Group A *(100% Accuracy)*

Data Group B *(0% Accuracy)*

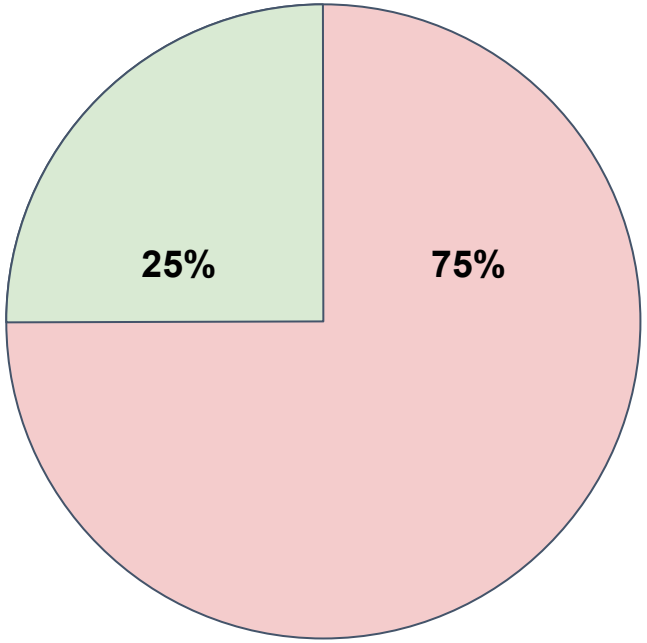
Training Split



IID Test Split



OOD Test Split

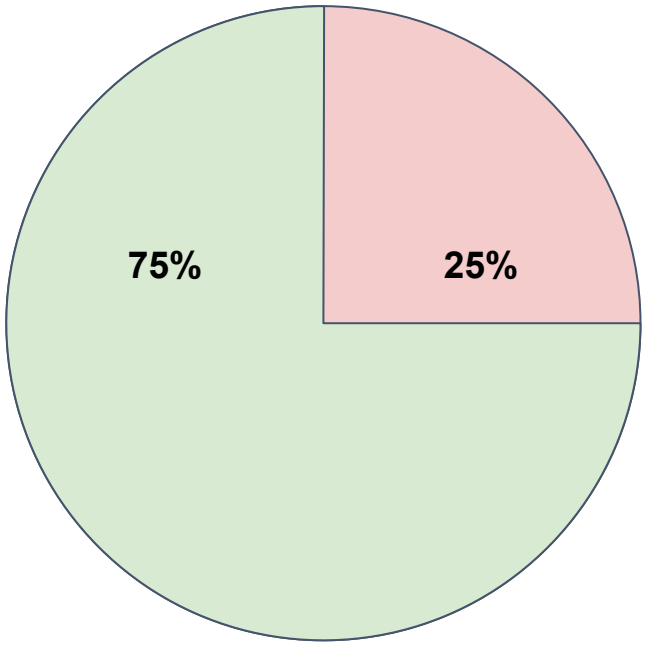


Problem: Bad Out-of-Distribution Performance

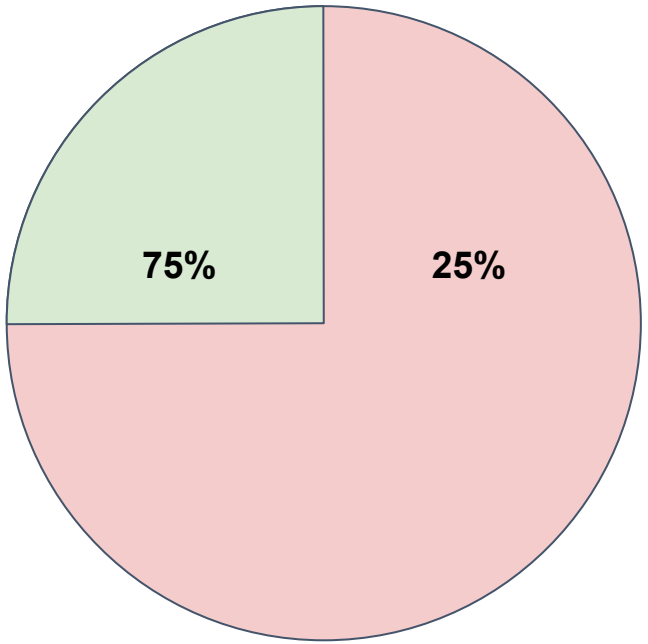
Data Group A *(100% Accuracy)*

Data Group B *(0% Accuracy)*

IID Test Split



OOD Test Split

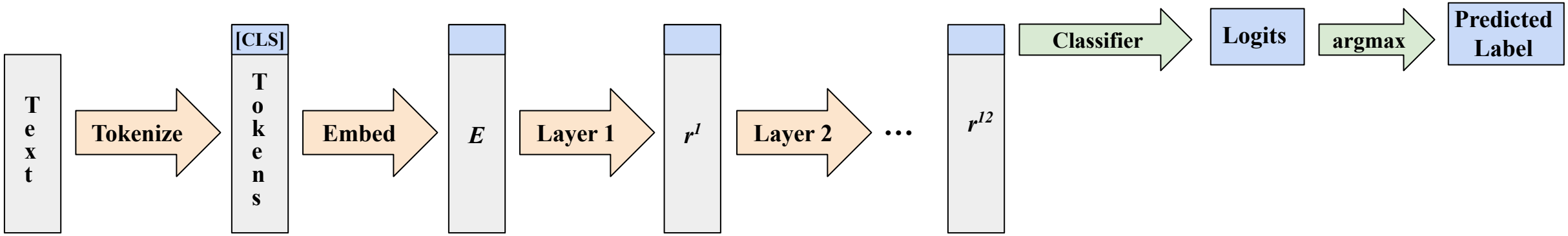


Accuracy on Training Split:
(0.75 * 100) + (0.25 * 0) = 75%

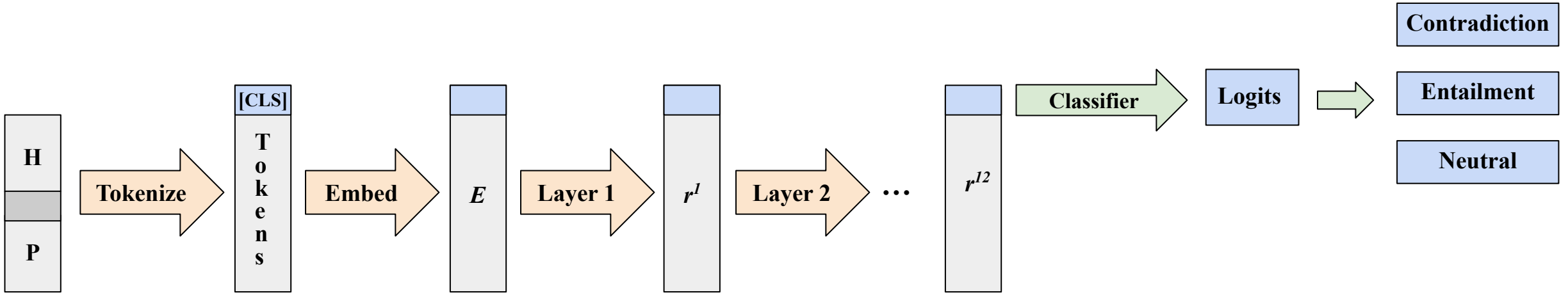
Accuracy on IID Test Split:
(0.75 * 100) + (0.25 * 0) = 75%

Accuracy on OOD Test Split:
(0.25 * 100) + (0.75 * 0) = 25%

Example: Text Classification



Example: Natural Language Inference



Problem: Natural Language Inference

Attribute	Label	Premise	Hypothesis
Negation Word $a = 1$	Contradiction $y = 0$	privatization assumes an end to the ponzi every generation saves for its own retirement.	privatization has ended the ponzi and generations no longer save for retirement.
	Entailment $y = 1$	only the right arm of one of the two transepts and the octagonal belltower [...] remain.	the left arms of the two transepts no longer remain.
	Neutral $y = 2$	uh yeah except i don't use them	i have never found the need to use them.

Natural Language Inference (NLI) Examples (Williams et al., 2018)

Problem: Natural Language Inference

Training

	Contradiction	Entailment	Neutral
Overall - $P(y)$	33.3%	33.4%	33.3%
No Negation - $P(y\|a=0)$	30.0%	35.2%	34.8%
Negation - $P(y\|a=1)$	76.1%	10.4%	13.6%

Test

No Negation Word			Negation Word		
Contradiction	Entailment	Neutral	Contradiction	Entailment	Neutral
82.0	**83.7**	**77.8**	**94.7**	**76.9**	**62.8**

Problem: Natural Language Inference

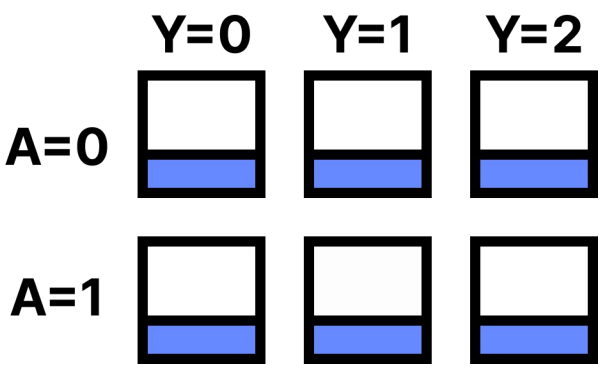
Training

	Contradiction	Entailment	Neutral
Overall - $P(y)$	33.3%	33.4%	33.3%
No Negation - $P(y\|a=0)$	30.0%	35.2%	34.8%
Negation - $P(y\|a=1)$	76.1%	10.4%	13.6%

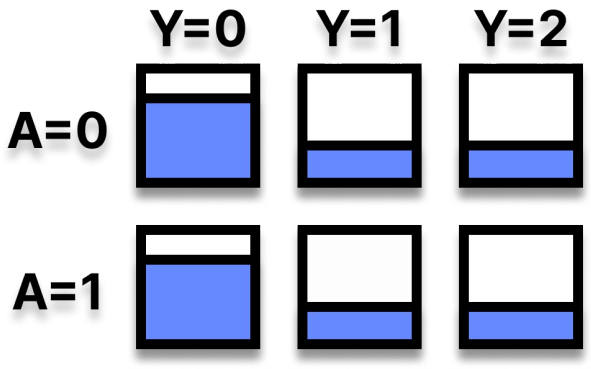
Test

No Negation Word			Negation Word		
Contradiction	Entailment	Neutral	Contradiction	Entailment	Neutral
82.0	**83.7**	**77.8**	**94.7**	**76.9**	**62.8**

Problem: Distribution Shifts

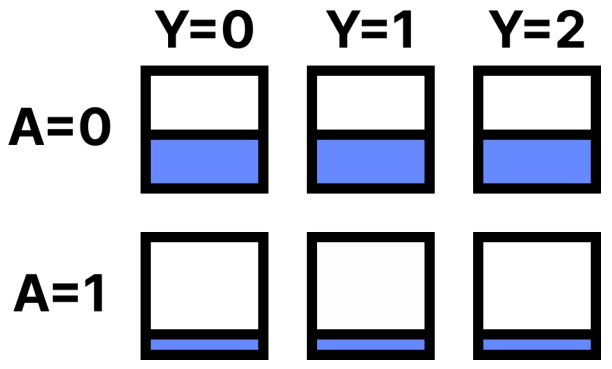


Balanced Dataset



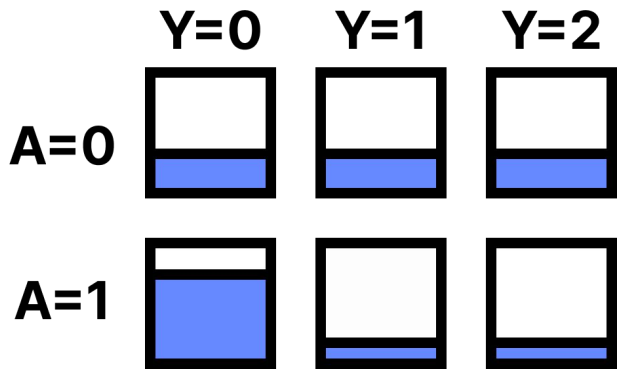
Label Imbalance

$$P(y_i) > P(y_j)$$



Attribute Imbalance

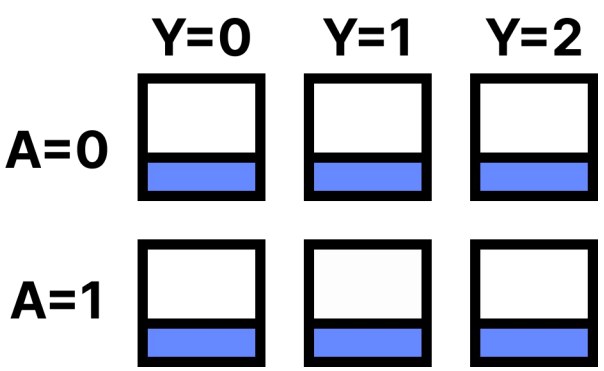
$$P(a_i) > P(a_j)$$



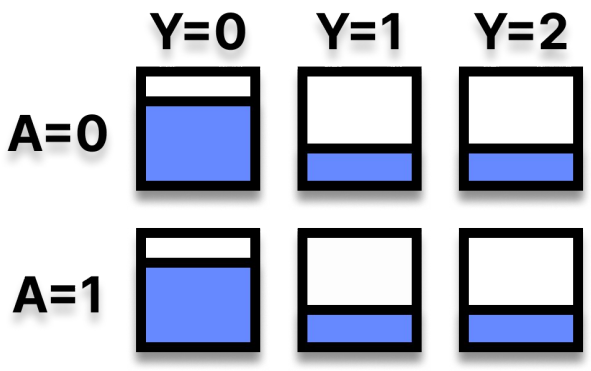
Spurious Correlation

$$P(y_i|a_i) > P(y_i|a_j)$$

Problem: Distribution Shifts

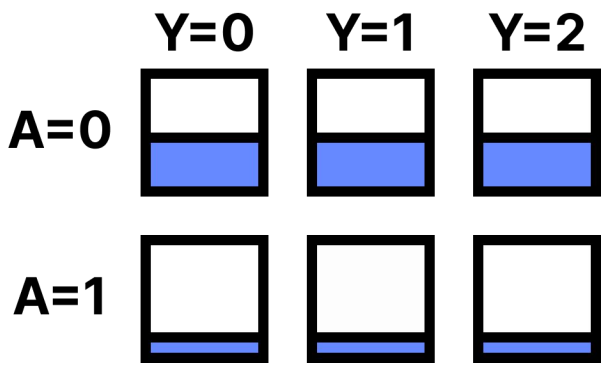


Balanced Dataset



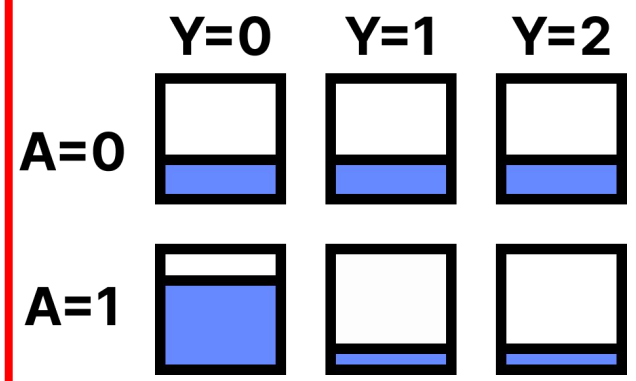
Label Imbalance

$P(y_i) > P(y_j)$



Attribute Imbalance

$P(a_i) > P(a_j)$



Spurious Correlation

$P(y_i|a_i) > P(y_i|a_j)$

Shortcuts

Existing Approaches

Empirical Risk Minimization

$$L(f_\theta(X), Y) = \sum_{i=1}^{m} \frac{1}{m} L(f_\theta(x_i), y_i),$$

where X is a batch of m inputs and Y are the corresponding outputs, f_θ represents the model, L is the loss function.

Empirical Risk Minimization

$$L(f_\theta(X), Y) = \sum_{i=1}^{m} w_i L(f_\theta(x_i), y_i),$$

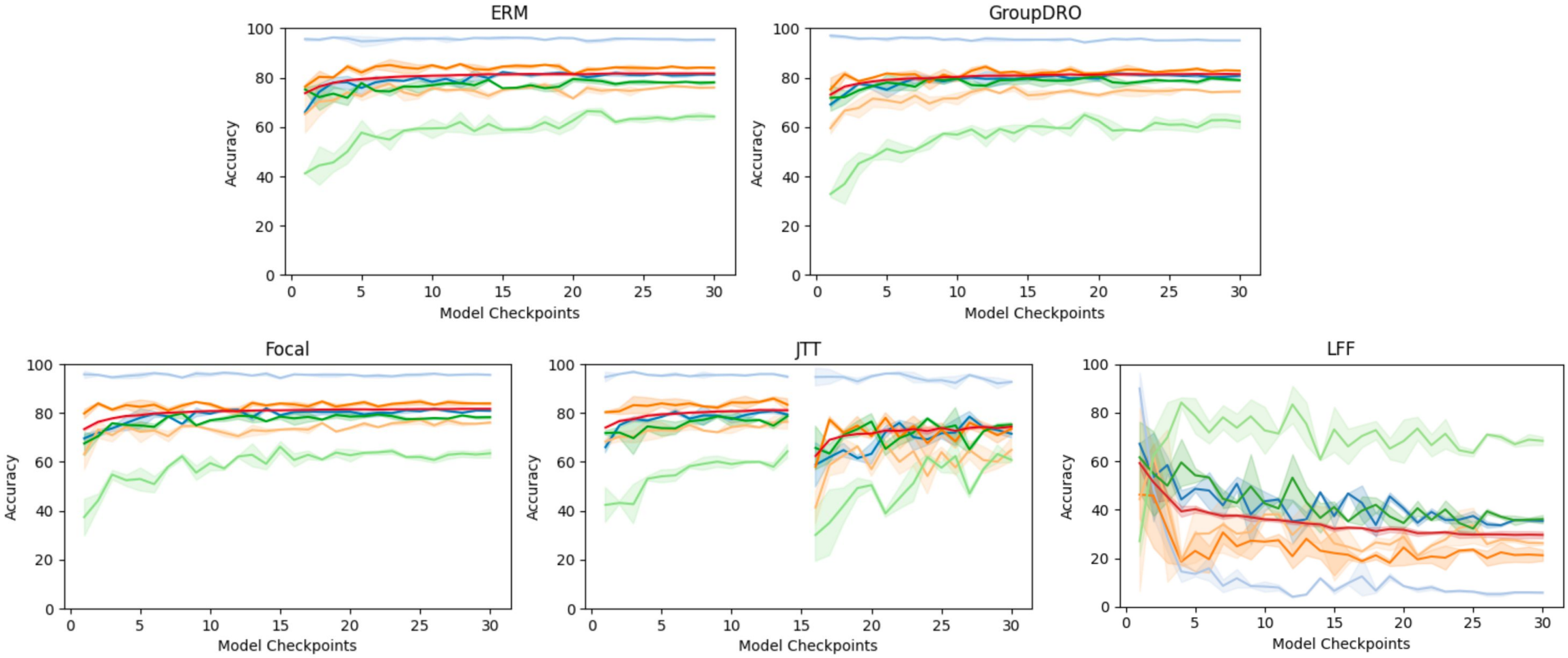
where X is a batch of m inputs and Y are the corresponding outputs, f_θ represents the model, L is the loss function.

Loss Function Based Approaches

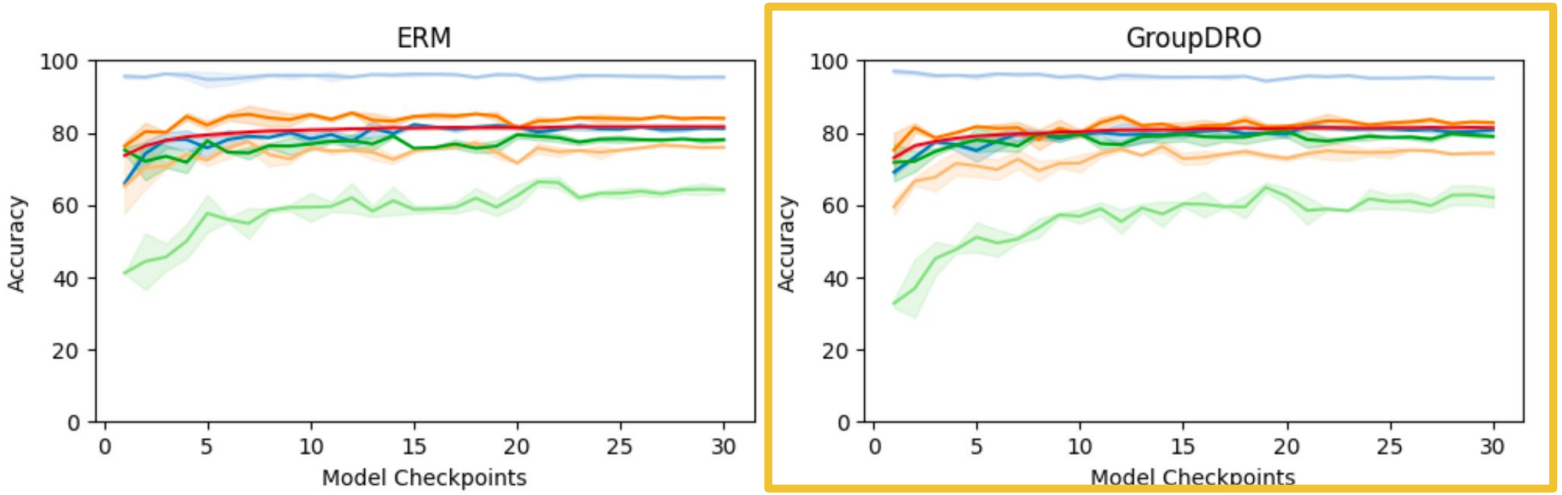
$$L(f_\theta(X), Y) = \sum_{i=1}^{m} w_i L(f_\theta(x_i), y_i),$$

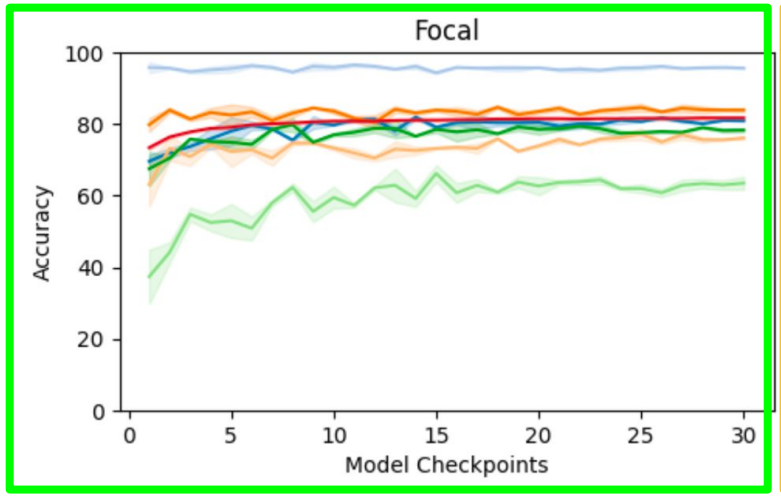
Method	Procedure
GroupDRO (Sagawa et al., 2020)	Weights the examples with the difficulty scores assigned to the groups based on the per group loss
Focal Loss (Lin et al., 2018)	Down-weights examples that are predicted with high confidence
JustTrain Twice (JTT) (Liu et al., 2021)	Up-weights examples that are misclassified by ERM
Learning From Failure (LfF) (Nam et al., 2020)	Weights the examples with difficulty scores assigned to each example based on the confidence of a second bias model for the example

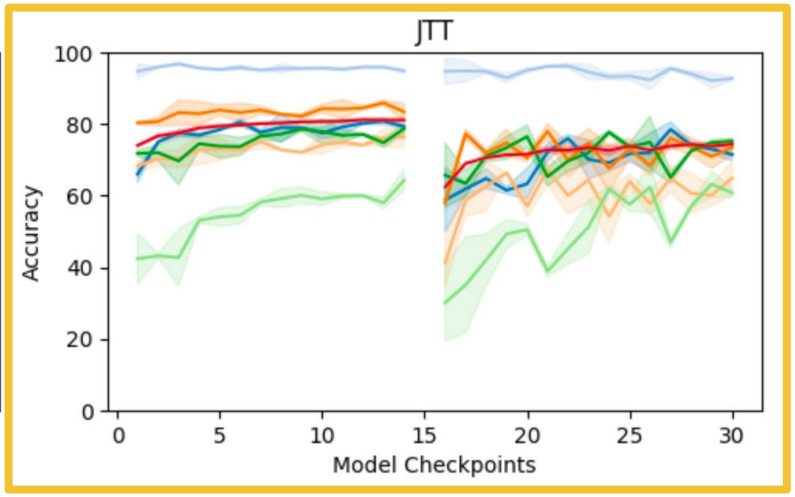
Training Dynamics: Natural Language Inference (MultiNLI)

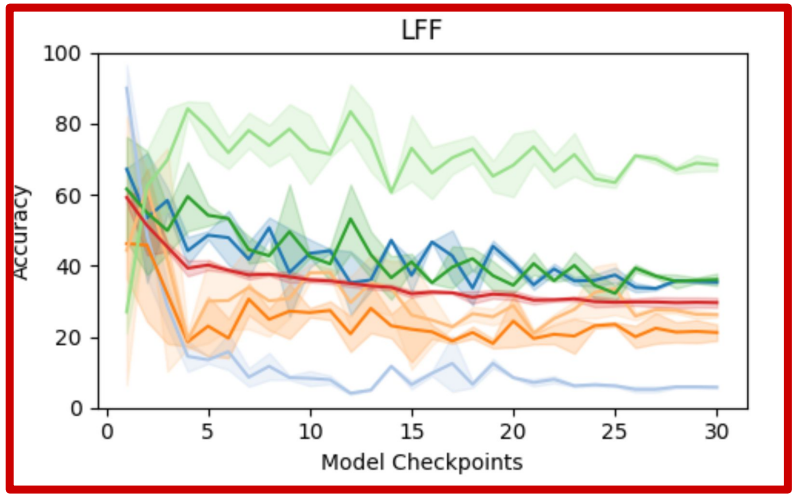


Training Dynamics: Natural Language Inference (MultiNLI)

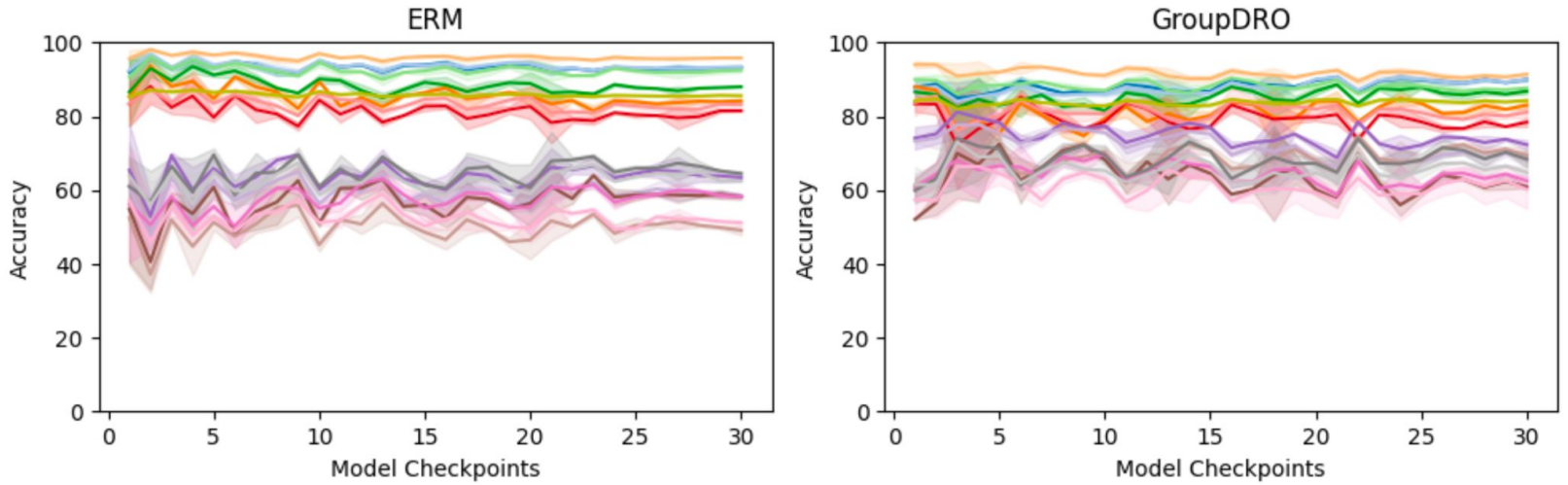


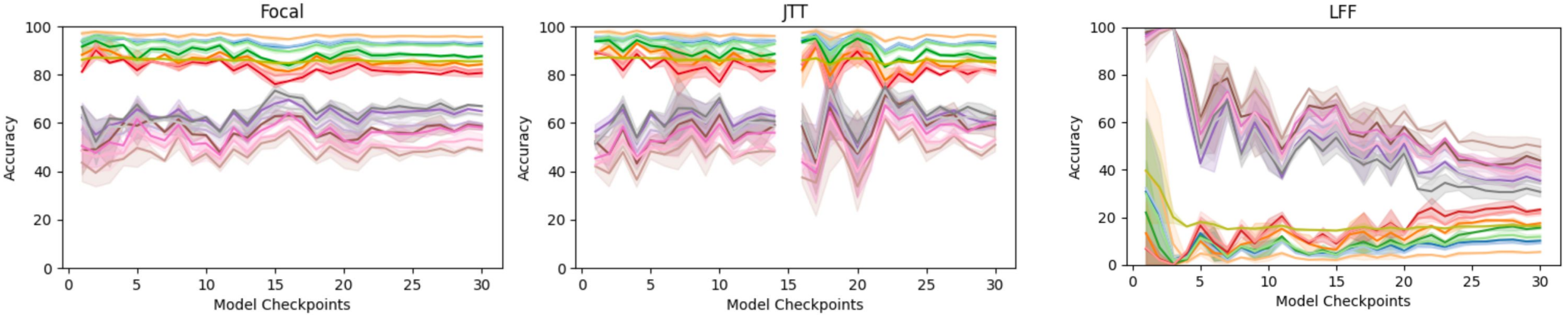




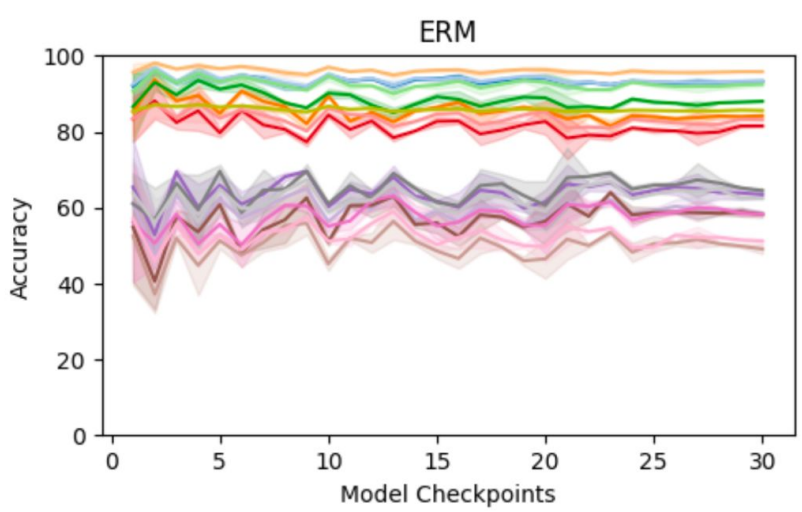


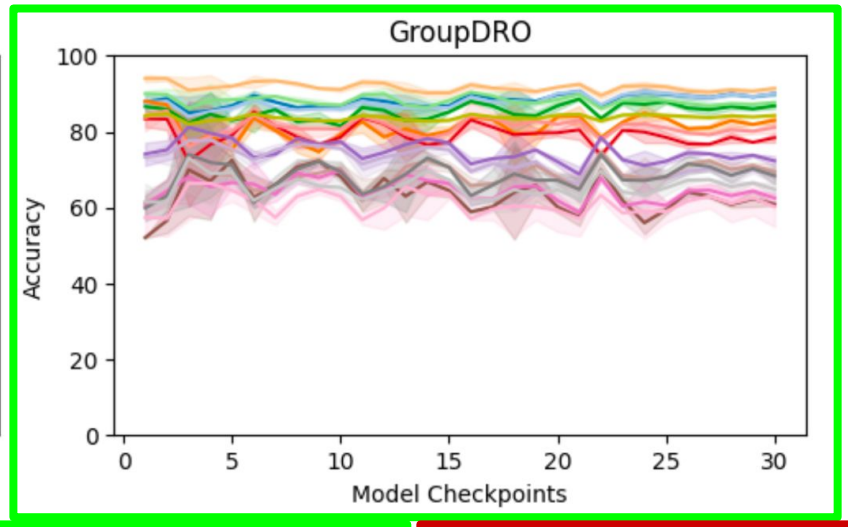
Training Dynamics: Toxicity Detection (CivilComments)

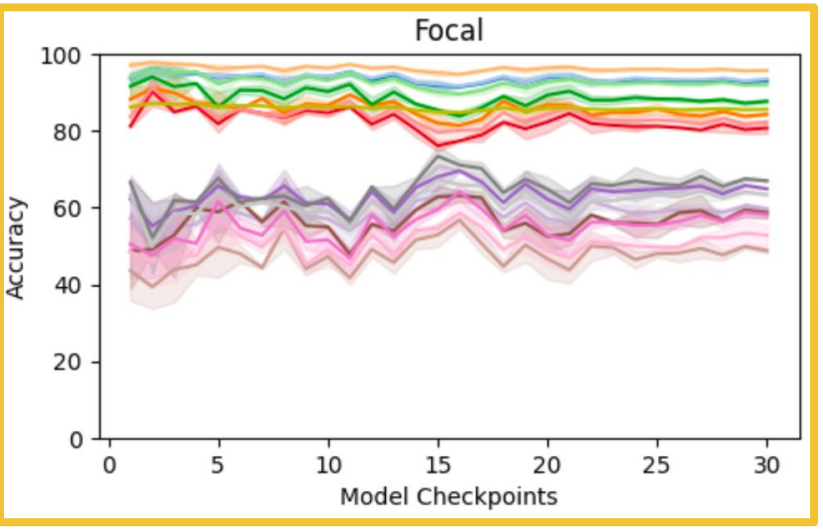


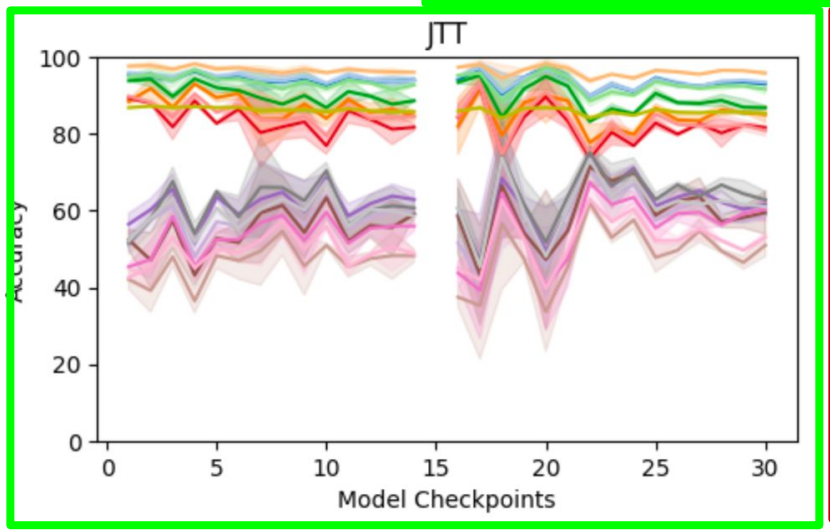


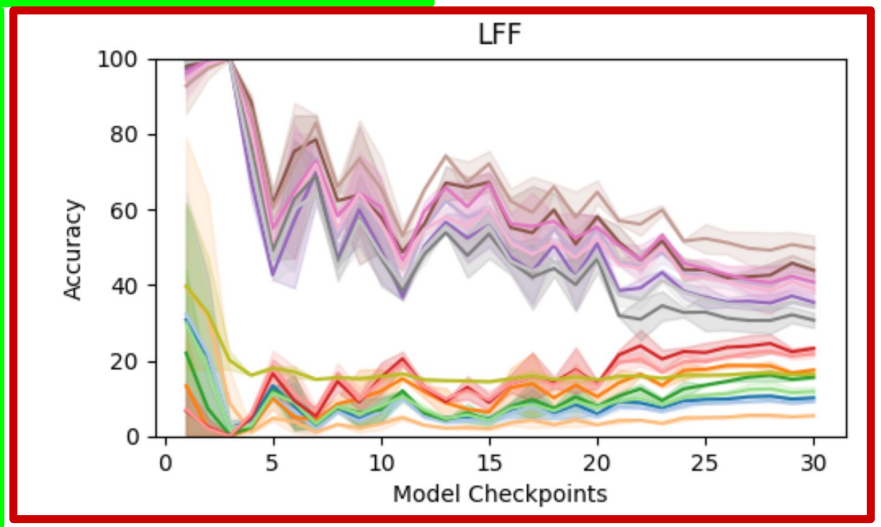
Training Dynamics: Toxicity Detection





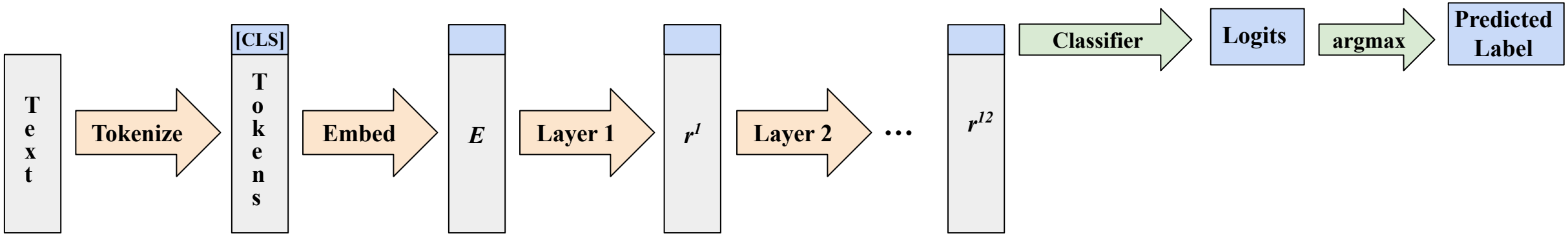




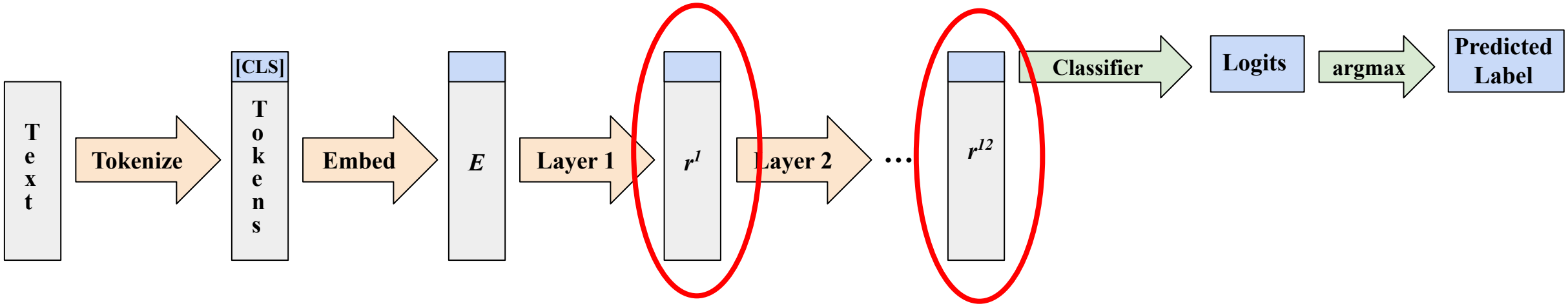


Analyzing Representations

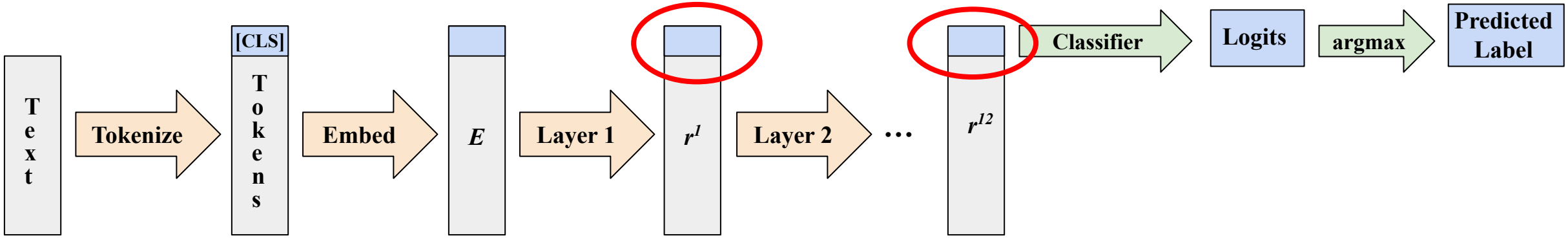
Analyzing Representations



Analyzing Representations

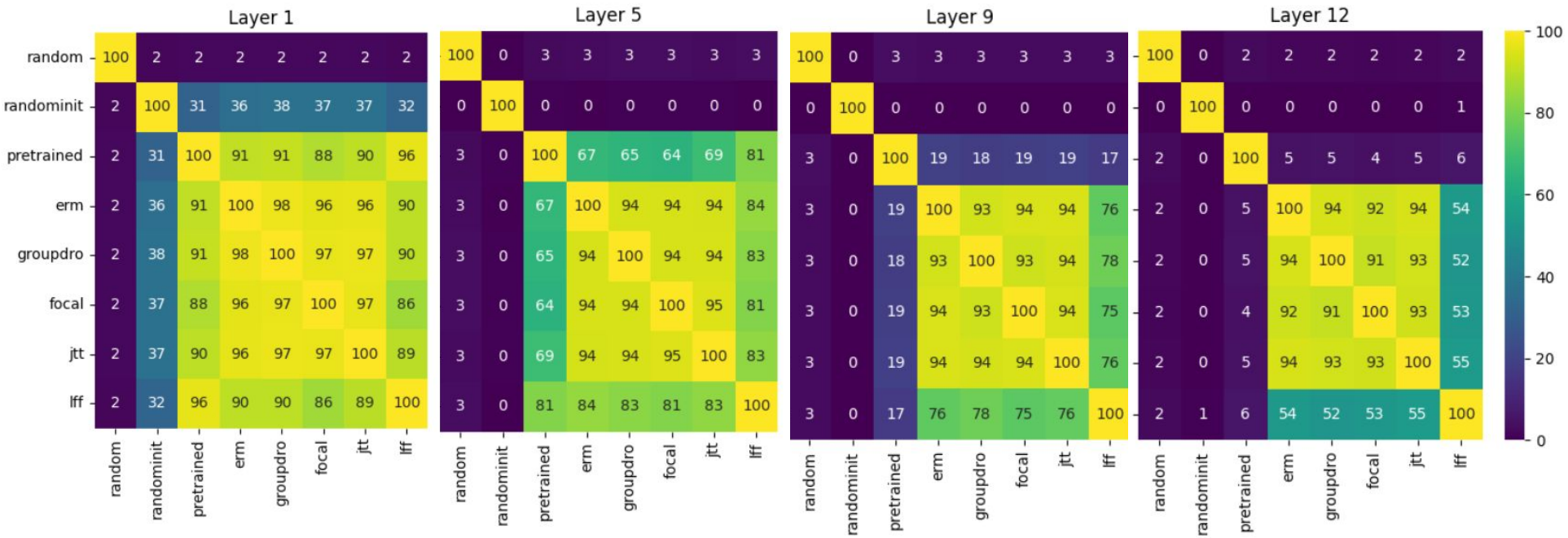


Analyzing Representations



Experiment I.
Representation Similarity

Representation Similarity (Centered Kernel Alignment)



Observations

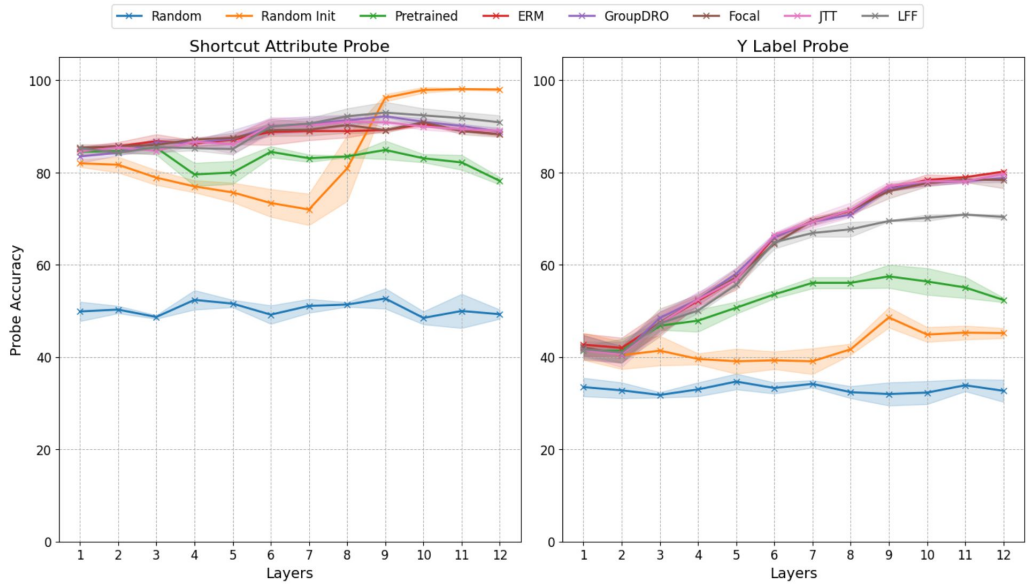
Divergence from pre-trained representations in later layers

Similarity between fine-tuning methods across all layers in MultiNLI

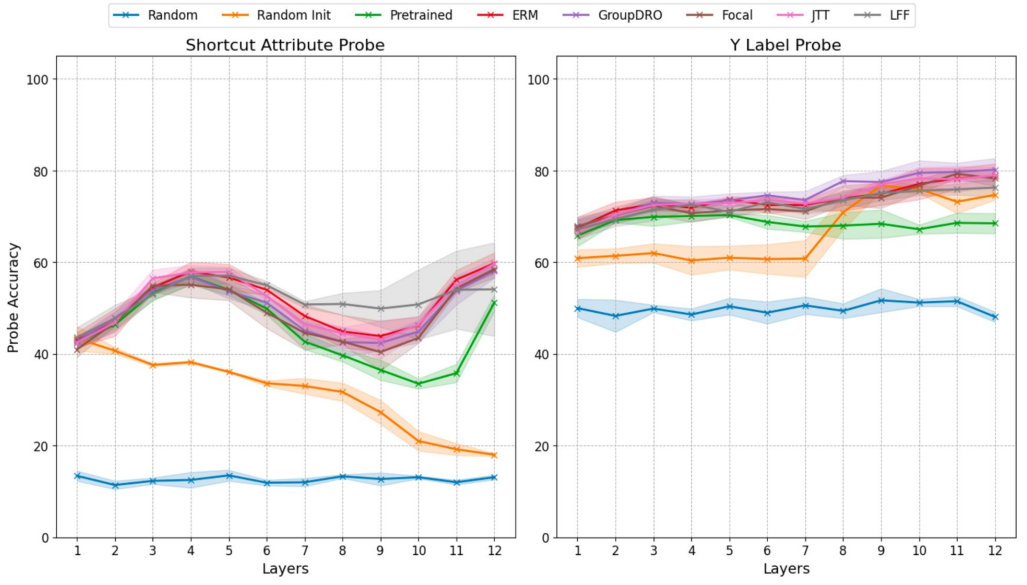
Periodicity in CivilComments

Experiment II.
Information Contained in the Representations

Probing Representations



MultiNLI



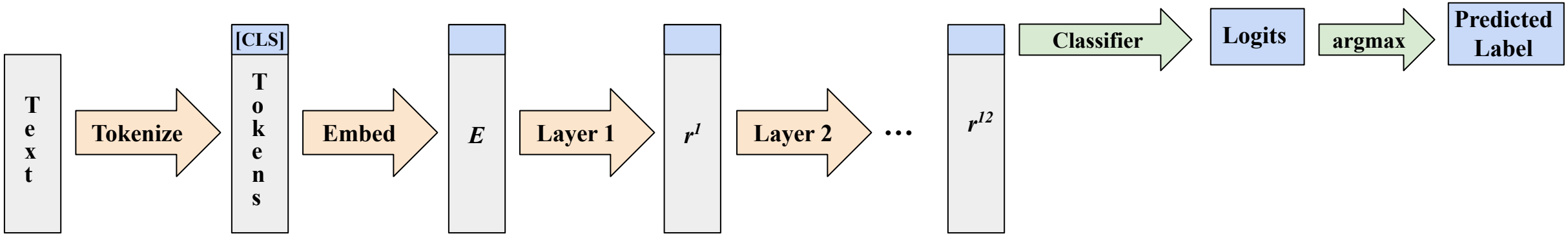
CivilComments

Observations

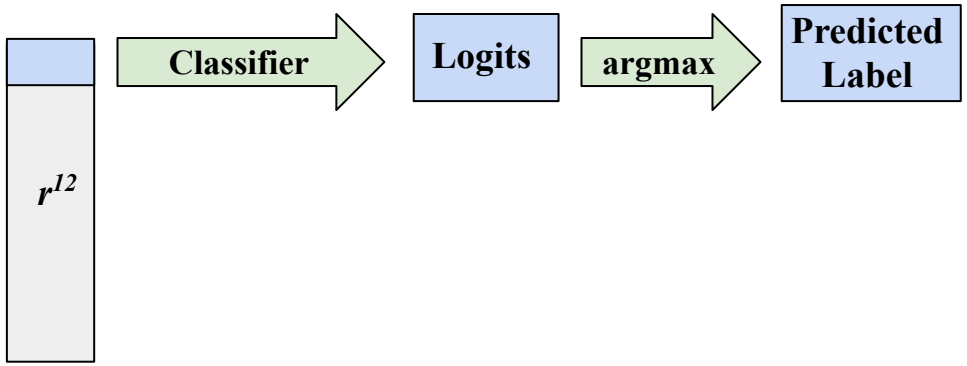
- Similarity between ERM and Loss-Function-Based approaches
- Difference in later layers between pre-trained and fine-tuned representations
- Randomly initialized models on binary probe targets
- Periodicity in CivilComments

Experiment III.
Information Used by the Model

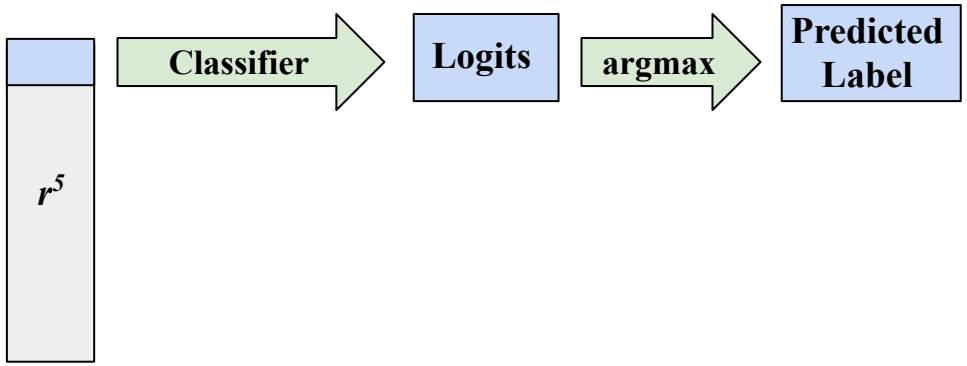
Logit Lens


Text
Tokenize
[CLS]
Tokens
Embed
E
Layer 1
r1
Layer 2
...
r12
Classifier
Logits
argmax
Predicted Label

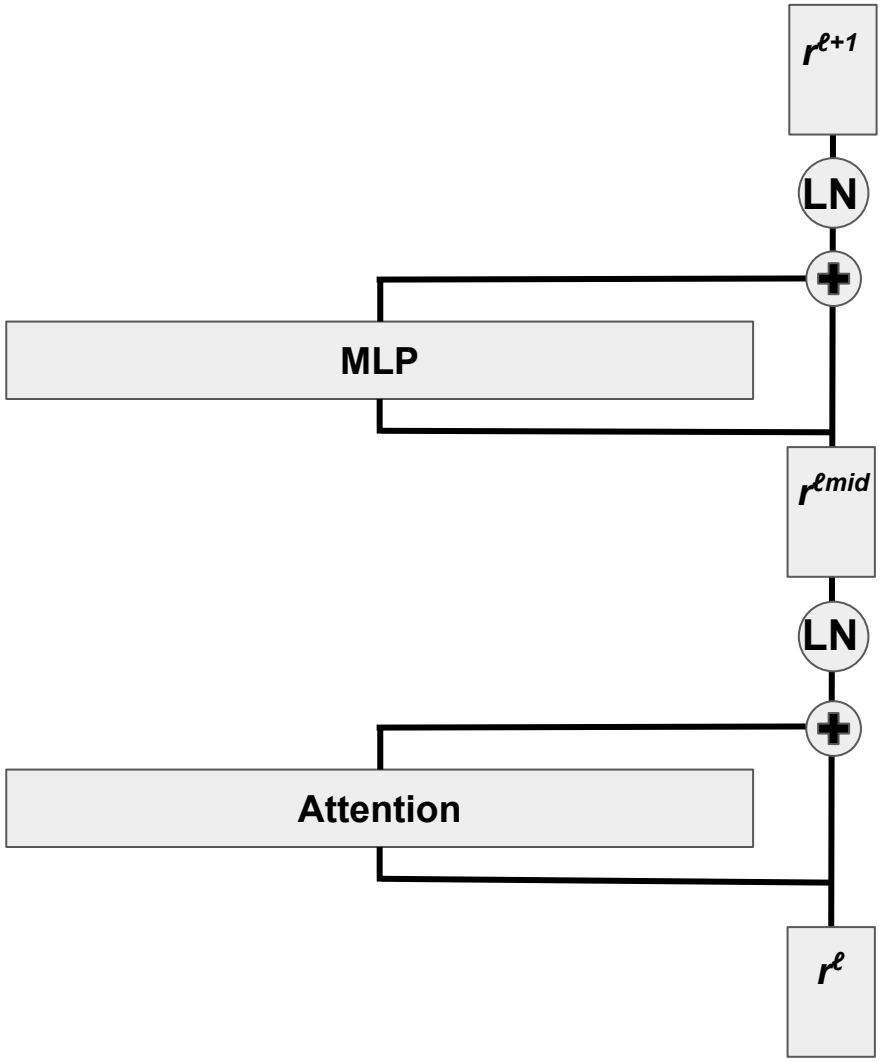
Logit Lens



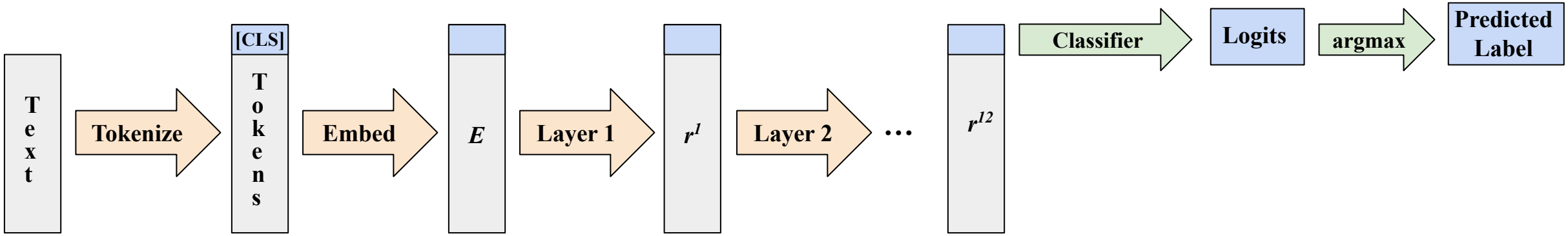
Logit Lens



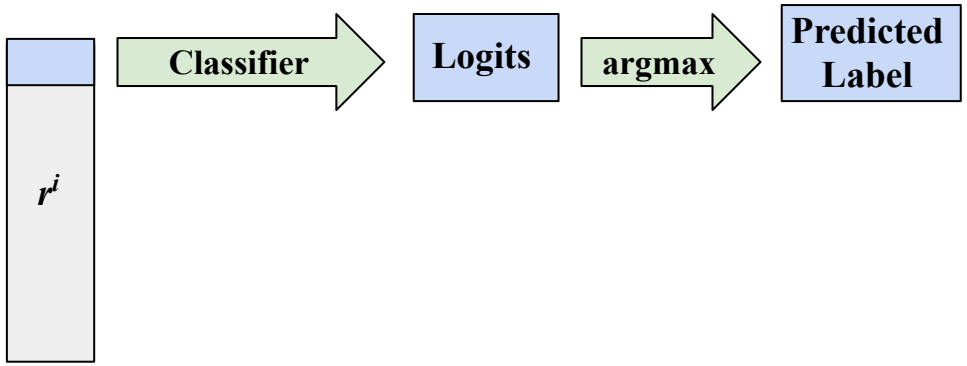
Logit Lens



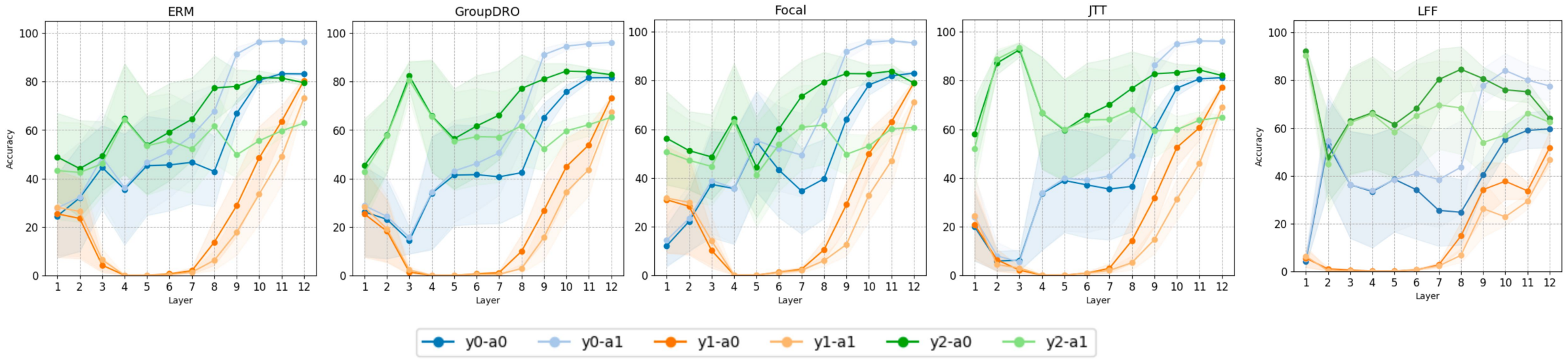
Logit Lens


Text
Tokenize
[CLS]
Tokens
Embed
E
Layer 1
r1
Layer 2
...
r12
Classifier
Logits
argmax
Predicted Label

Logit Lens



Logit Lens



Observations

Accuracy for different labels improve at different layers

Divergence across attributes in mid-layers

A New Perspective

A New Perspective: Interpretability-Based Approach

Competing Rules Hypothesis (CRH)

- We can understand the model predictions as a ***competition*** between an **intended rule** and a **shortcut rule**.[3]

Shifting the Representations (REPRSHIFT)

- We can then systematically shift the representations to suppress the shortcut rule, so that the intended rule dominates the predictions.

[3] Ortu et al., (2024)

Competing Rules Hypothesis (CRH)

A New Perspective: Interpretability-Based Approach

Competing Rules Hypothesis (CRH)

- We can understand the model predictions as a ***competition*** between an **intended rule** and a **shortcut rule**.[3]

Algorithm Intended Rule (NLI example)

Input: Premise, Hypothesis

if *Premise contradicts hypothesis* **then**
  Increase contradiction logit, decrease others

else if *Premise entails hypothesis* **then**
  Increase entailment logit, decrease others

else
  Increase neutral logit, decrease others

A New Perspective: Interpretability-Based Approach

Competing Rules Hypothesis (CRH)

- We can understand the model predictions as a ***competition*** between an **intended rule** and a **shortcut rule**.[3]

Algorithm Shortcut Rule (NLI example)

Input: Hypothesis

if *Hypothesis contains a negation word* **then**

  Increase contradiction logit, decrease others

A New Perspective: Interpretability-Based Approach

Competing Rules Hypothesis (CRH)

- We can understand the model predictions as a ***competition*** between an **intended rule** and a **shortcut rule**.[3]

Algorithm Intended Rule (general)

Input: x

Influence the logits to predict $P_{\text{train}}(y|x_R)$, where x_R are robust features

---

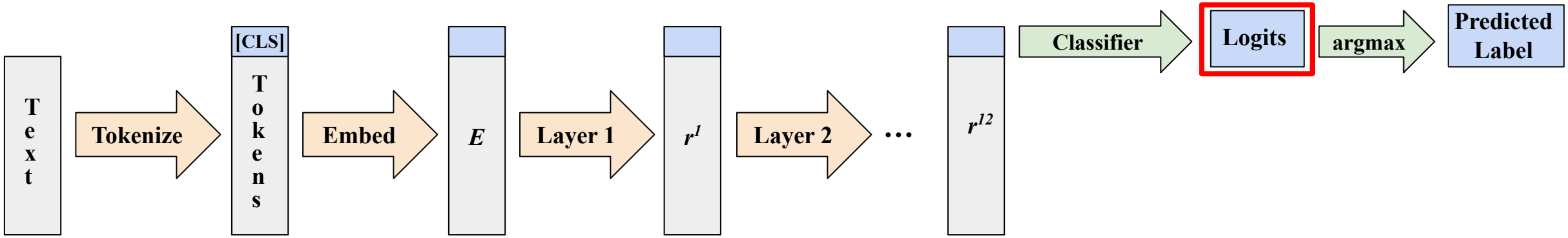
Algorithm Shortcut Rule (general)

Input: x

if x *contains attribute* $a_{shortcut}$ **then**

  Influence the logits to predict $P_{\text{train}}(y|a_{\text{shortcut}})$

A New Perspective: Interpretability-Based Approach



Suggestive Evidence for CRH

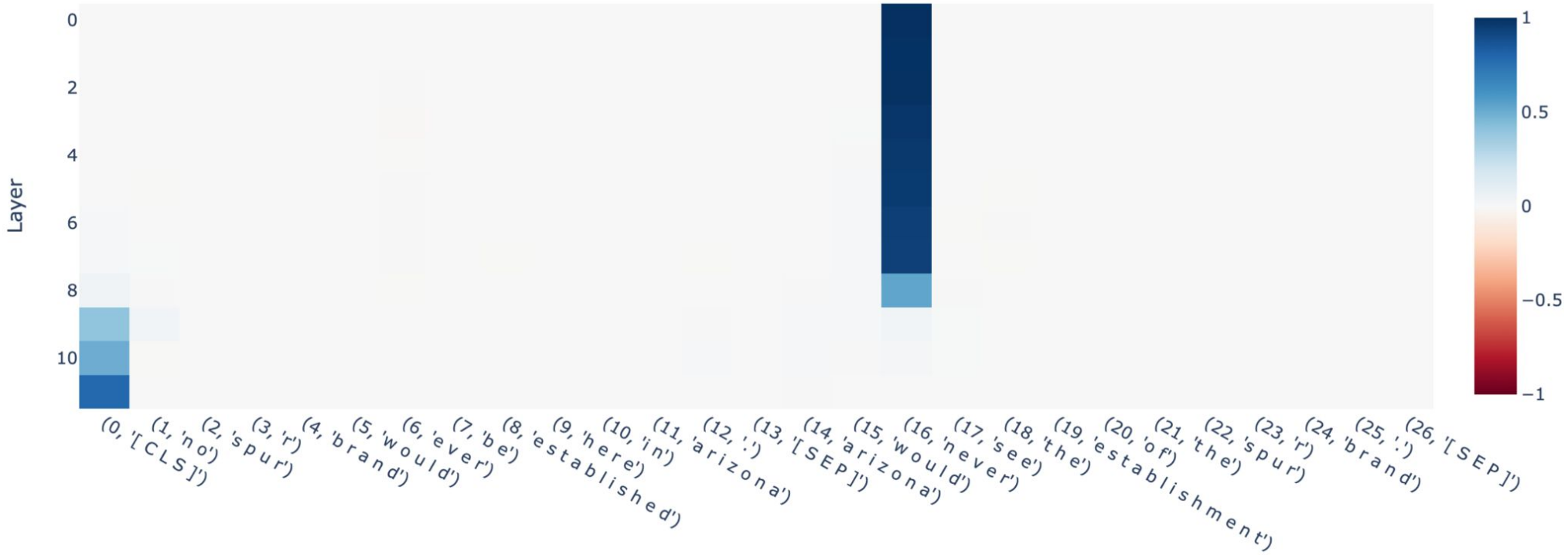
1. Constructive Interference

	No Shortcut Attribute			Shortcut Attribute		
	Contradiction	Entailment	Neutral	Contradiction	Entailment	Neutral
Contradiction Logit	**2.91**	-1.94	-1.20	**4.03**	-1.29	0.02
Entailment Logit	-2.26	**2.79**	-1.26	-3.02	**2.20**	-1.80
Neutral Logit	-0.90	-0.65	**2.52**	-1.34	-0.63	**1.75**

2. Destructive Interference

	No Shortcut Attribute			Shortcut Attribute		
	Contradiction	Entailment	Neutral	Contradiction	Entailment	Neutral
Contradiction Logit	**2.91**	-1.94	-1.20	**4.03**	-1.29	0.02
Entailment Logit	-2.26	**2.79**	-1.26	-3.02	**2.20**	-1.80
Neutral Logit	-0.90	-0.65	**2.52**	-1.34	-0.63	**1.75**

3. Narrow Channels



Representation Shift (REPRSHIFT)

[4] Bau et al. (2020) and Meng et al. (2023)

REPRSHIFT: Model Algebra

We want to have a model with only INTENDED RULE

We have a model with $\text{INTENDED RULE} + \text{SHORTCUT RULE}$

We compute $\text{INVERSE SHORTCUT RULE} = -\ \text{SHORTCUT RULE}$

We add inverse shortcut rule to our model and get a model with

$$\text{INTENDED RULE} + \text{SHORTCUT RULE} + \text{INVERSE SHORTCUT RULE}$$

This is equal to

$$\text{INTENDED RULE} + \text{SHORTCUT RULE} + (-\ \text{SHORTCUT RULE}\ )$$

Shortcut rules cancel out and we get a model with

$$\text{INTENDED RULE}$$

REPRSHIFT: Inverse Shortcut Rule

Algorithm Shortcut Rule

Input: Hypothesis

if *Hypothesis contains a negation word* **then**

  Increase contradiction logit, decrease others

Algorithm Inverse Shortcut Rule

Input: Hypothesis

if *Hypothesis contains a negation word* **then**

  Decrease contradiction logit, increase others

REPRSHIFT: Inverse Shortcut Rule

Algorithm Shortcut Rule (general)

Input: x

if x *contains attribute* $a_{shortcut}$ **then**

  Influence the logits to predict $P_{\text{train}}(y|a_{\text{shortcut}})$

Algorithm Inverse Shortcut Rule (general)

Input: x

if x *contains attribute* $a_{shortcut}$ **then**

  Influence the logits to predict $(1 - P_{\text{train}}(y|a_{\text{shortcut}}))$

REPRSHIFT: Model Algebra

We want to have a model with only INTENDED RULE

We have a model with $\text{INTENDED RULE} + \text{SHORTCUT RULE}$

We compute $\text{INVERSE SHORTCUT RULE} = -\ \text{SHORTCUT RULE}$

We add inverse shortcut rule to our model and get a model with

$$\text{INTENDED RULE} + \text{SHORTCUT RULE} + \text{INVERSE SHORTCUT RULE}$$

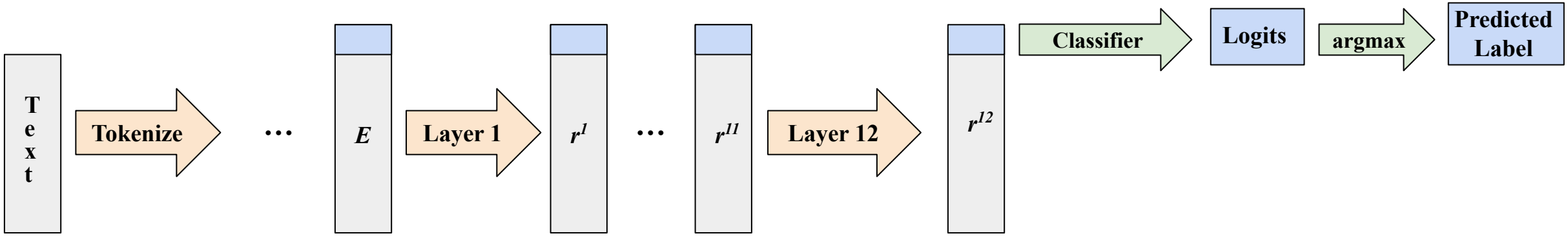
This is equal to

$$\text{INTENDED RULE} + \text{SHORTCUT RULE} + (-\ \text{SHORTCUT RULE}\ )$$

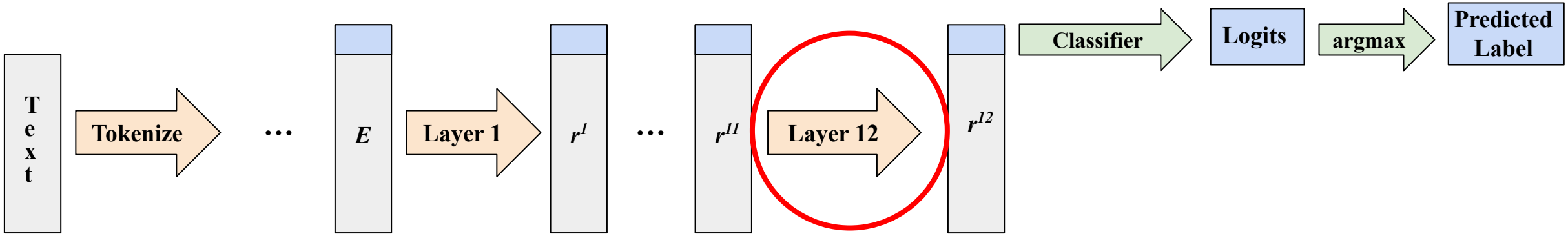
Shortcut rules cancel out and we get a model with

$$\text{INTENDED RULE}$$

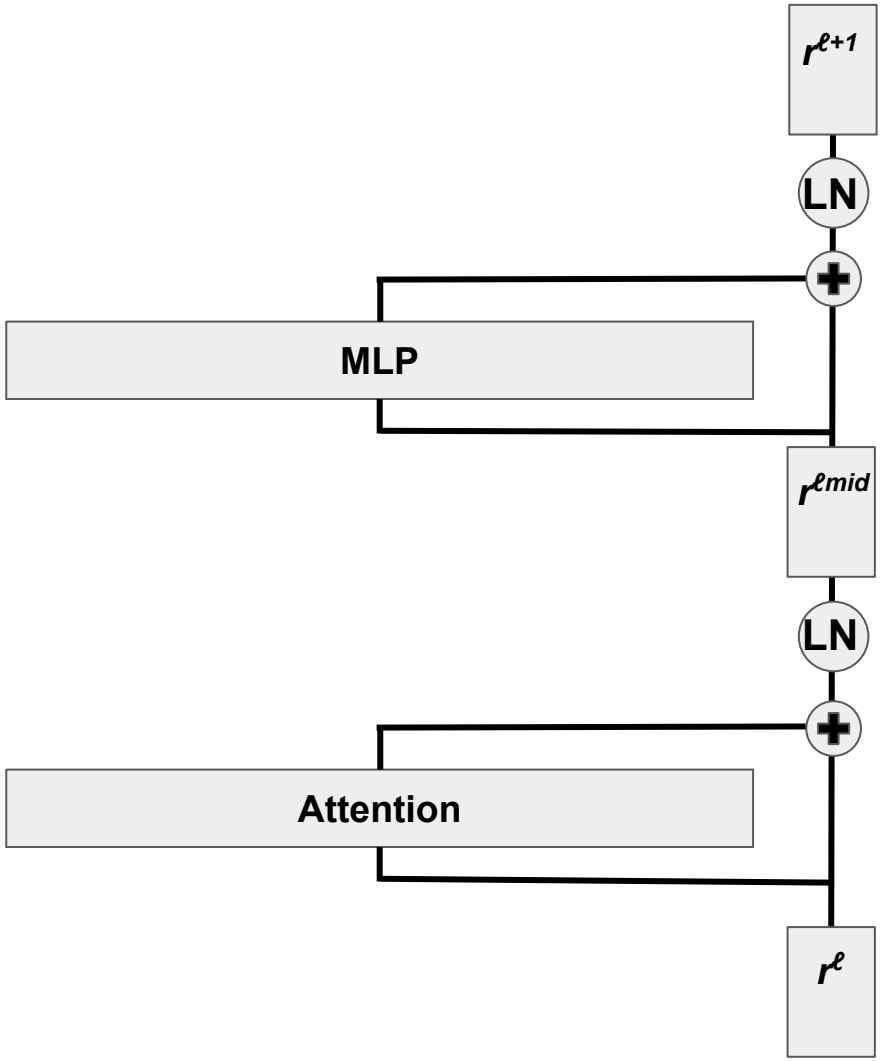
REPRSHIFT


Text
Tokenize
...
E
Layer 1
r^1
...
r^{11}
Layer 12
r^{12}
Classifier
Logits
argmax
Predicted Label

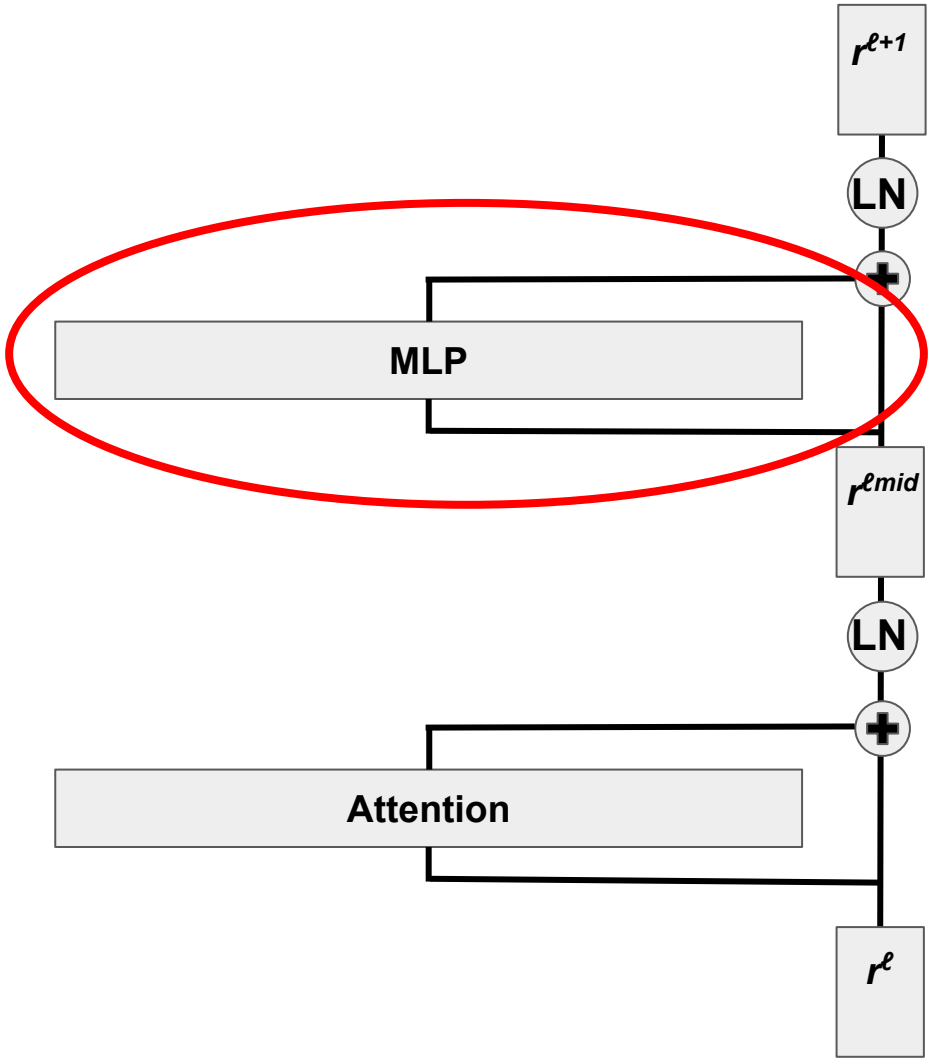
REPRSHIFT


Text
Tokenize
...
E
Layer 1
r1
...
r11
Layer 12
r12
Classifier
Logits
argmax
Predicted Label

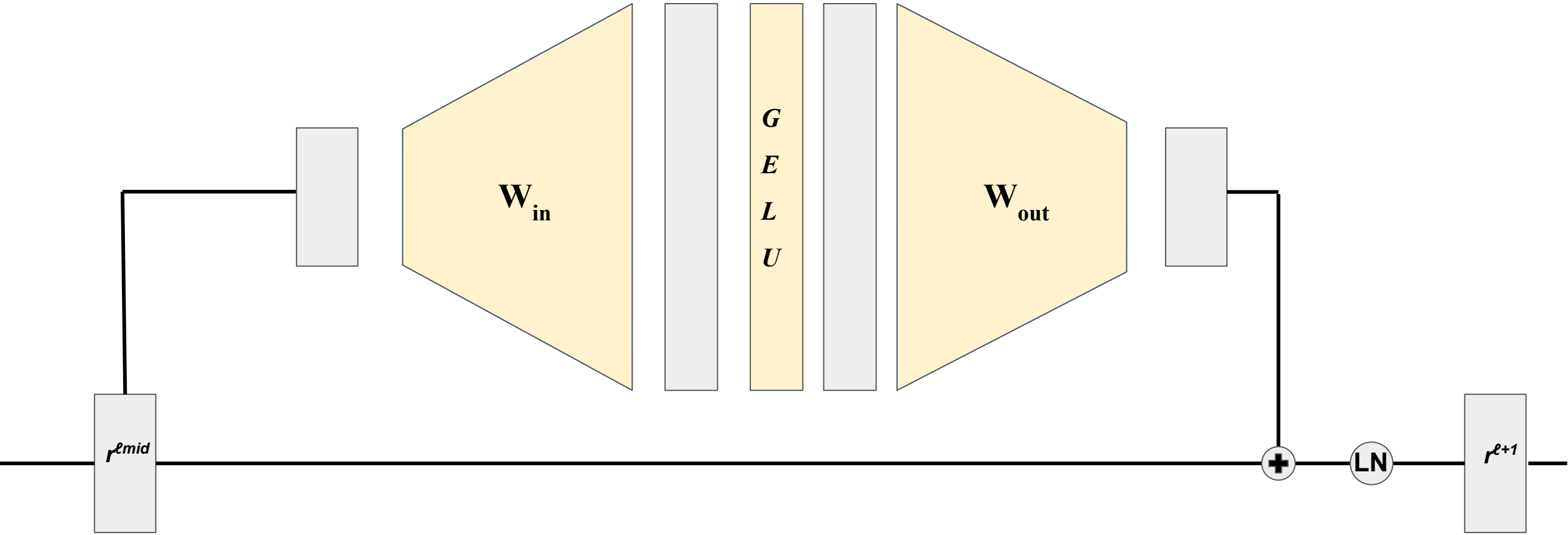
REPRSHIFT



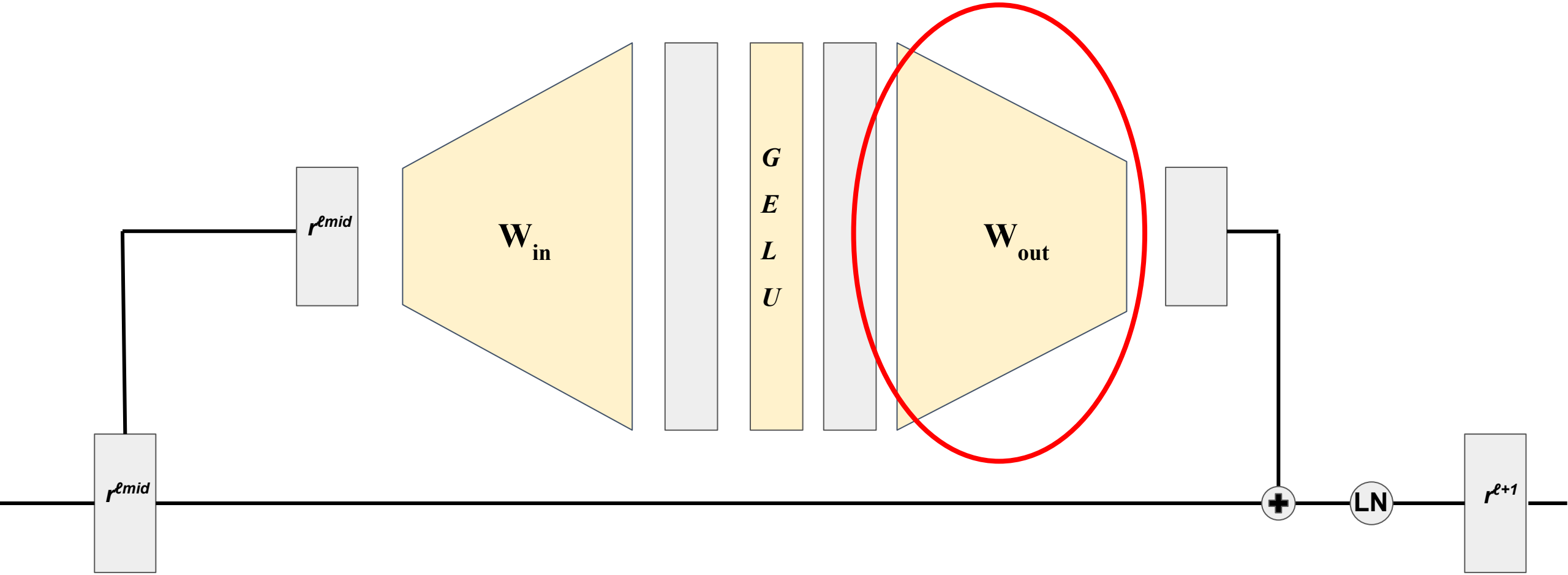
REPRSHIFT



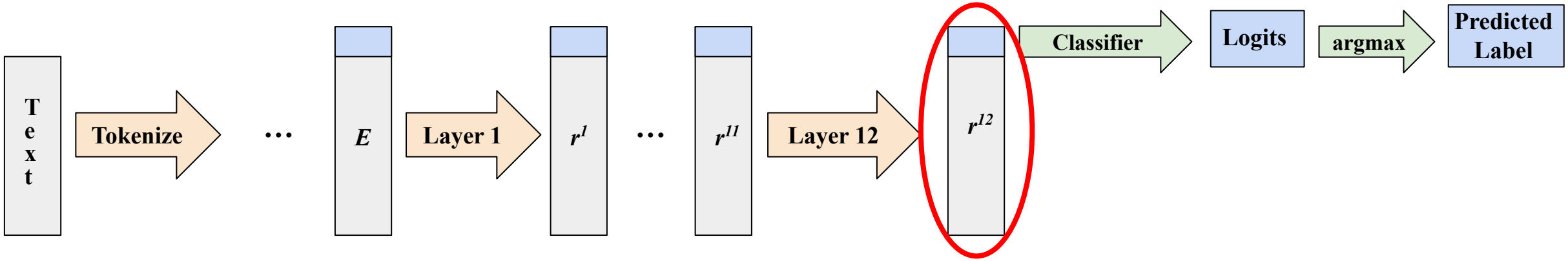
REPRSHIFT: MLP Layer



REPRSHIFT: MLP Layer



REPRSHIFT


Text
Tokenize
...
E
Layer 1
r^1
...
r^{11}
Layer 12
r^{12}
Classifier
Logits
argmax
Predicted Label

REPRSHIFT: Steps

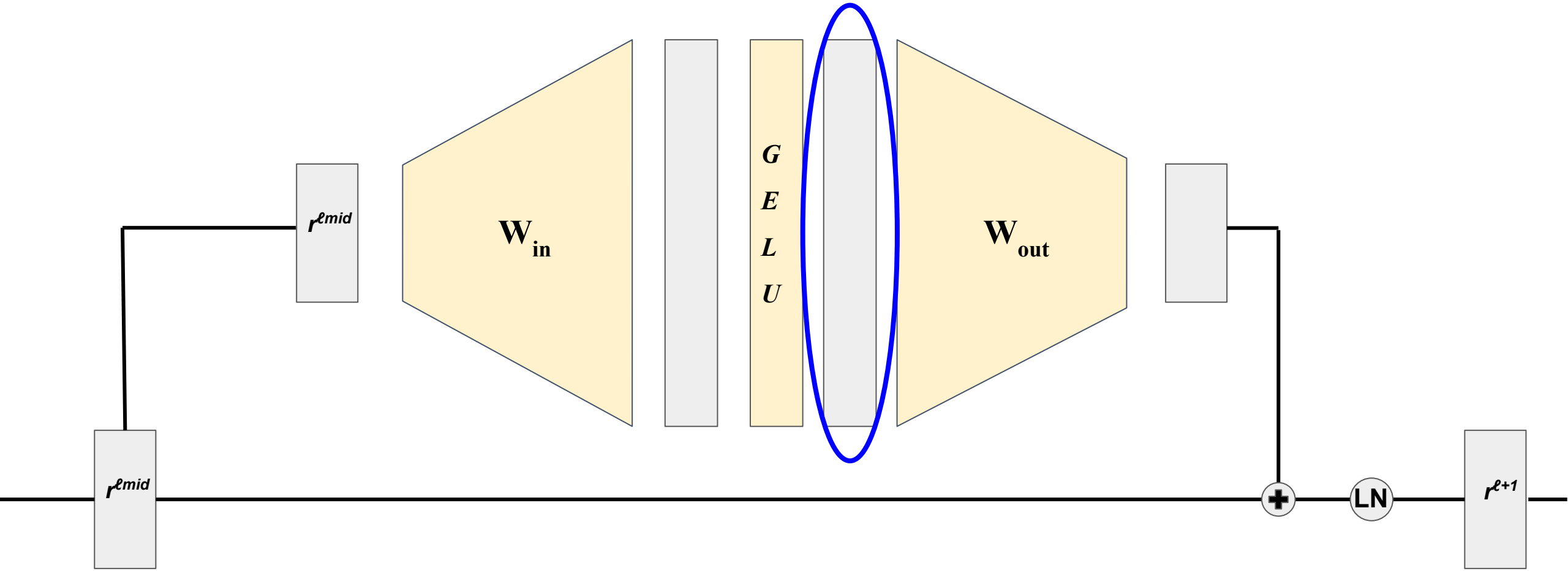
1. We compute an internal representation, k_{shortcut}
2. We compute a logit shift vector, $v_{\text{logitshift}}$
3. We use this logit shift vector to compute a representation shift vector, $v_{\text{reprshift}}$
4. Finally, we modify a single MLP weight matrix W inside our model to implement the inverse shortcut rule[4]:

if k_{shortcut}, **then** $v_{\text{reprshift}}$ or, equivalently, $k_{\text{shortcut}} \cdot W = v_{\text{reprshift}}$

Step 1: Computing $k_{shortcut}$

Algorithm Inverse Shortcut Rule

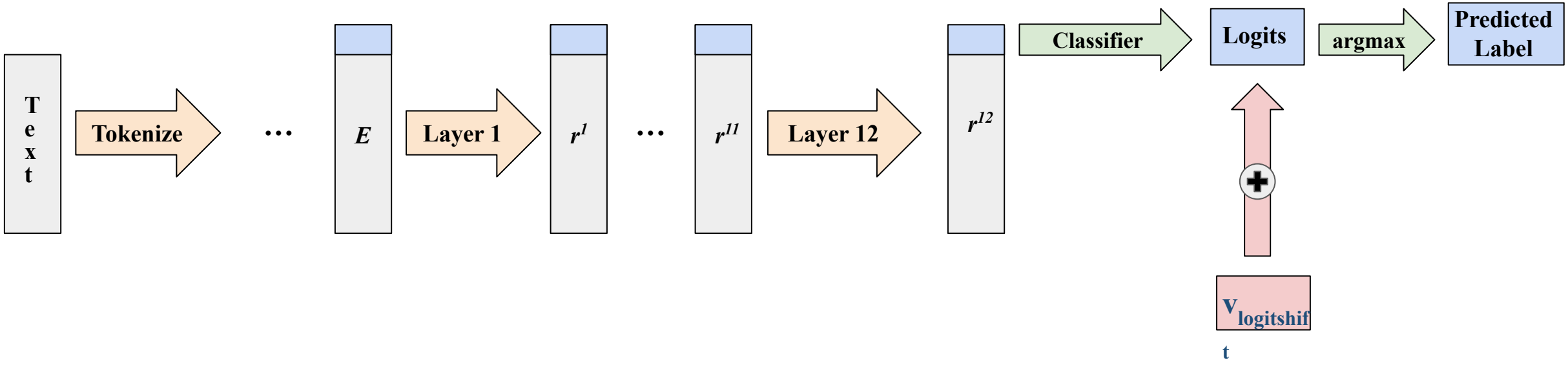
Input: Hypothesis
if *Hypothesis contains a negation word* **then**
  Decrease contradiction logit, increase others



Step 2: Computing $v_{logitshift}$

Algorithm Inverse Shortcut Rule

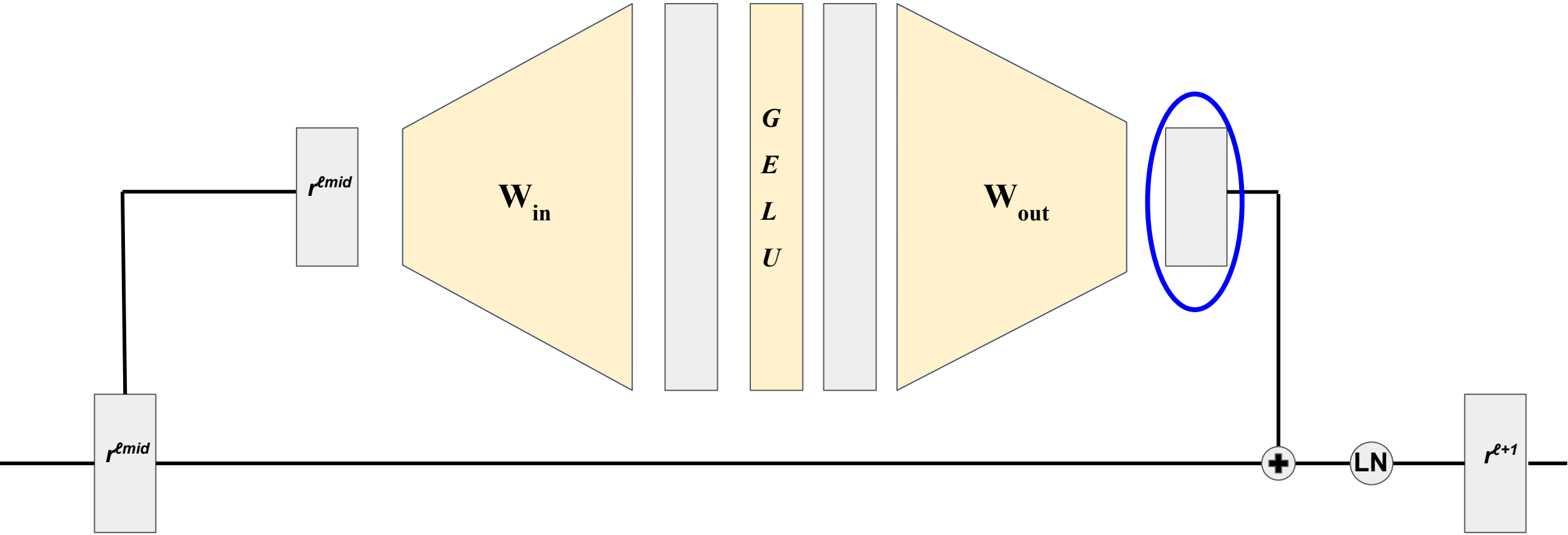
Input: Hypothesis
if *Hypothesis contains a negation word* **then**
  Decrease contradiction logit, increase others



Step 3: Computing $v_{reprshift}$

Algorithm Inverse Shortcut Rule

Input: Hypothesis
if *Hypothesis contains a negation word* **then**
  Decrease contradiction logit, increase others



Step 4: Modifying the Weights

Algorithm Inverse Shortcut Rule

Input: Hypothesis
if *Hypothesis contains a negation word* **then**
  Decrease contradiction logit, increase others

if k_{shortcut}, **then** $v_{\text{reprshift}}$ or, equivalently, $k_{\text{shortcut}} \cdot W = v_{\text{reprshift}}$

[4] Bau et al. (2020) and Meng et al. (2023)

Evaluating REPRSHIFT

	ERM	GroupDRO	Focal	JTT	LFF	REPRSHIFT
Acc.	81.2 (0.6)	81.1 (0.2)	81.1 (0.3)	80.7 (0.1)	59.0 (2.7)	79.8 (0.2)
WGA	64.6 (1.1)	64.3 (1.8)	65.6 (2.2)	64.3 (2.3)	46.1 (13.4)	**72.9 (3.1)**
EWA	79.8 (0.1)	79.3 (0.2)	79.6 (0.4)	78.9 (0.0)	58.1 (2.9)	79.5 (0.1)
y=0, a=0	79.2 (1.2)	80.1 (0.5)	79.2 (0.6)	78.1 (0.4)	58.3 (16.5)	77.8 (0.3)
y=0, a=1	94.5 (0.8)	94.2 (0.2)	94.7 (0.5)	94.8 (0.4)	76.6 (13.1)	88.9 (1.7)
y=1, a=0	83.2 (0.9)	81.1 (0.9)	82.0 (1.4)	81.9 (0.8)	52.1 (16.6)	83.0 (0.7)
y=1, a=1	77.0 (1.3)	73.6 (1.7)	75.0 (0.3)	74.8 (1.0)	46.1 (13.4)	77.3 (1.4)
y=2, a=0	79.3 (0.5)	80.3 (1.4)	80.2 (0.5)	80.1 (0.4)	64.0 (6.2)	77.0 (0.2)
y=2, a=1	64.6 (1.1)	64.3 (1.8)	65.6 (2.2)	64.3 (2.3)	62.0 (10.7)	72.9 (3.1)

REPRSHIFT Evaluation on NLI.[5]

[5] Accuracy (Acc.), Worst-Group Accuracy (WGA), Equally Weighted Accuracy(EWA)

Thank you